Chapter 12

BY :

VISHU GOYAL

ETISHA JAIN

LINUX 12.1 Understanding cron and at

- **Cron** is a daemon that triggers jobs on a regular basis.
- It works with different configuration files that specify when a job should be started.
- Use it for regular re-ocurring jobs,like **backup jobs.**
- **at** is used for tasks that need to be started once .
- **Systemd timers** provide a new alternative to Cron(not as precise as cron)

Linux 12.2 Understanding cron scheduling options

• **Rsyslog** is a king of legacy logging service,and kind of not becox systemd journald is also taking care of logging with our syslog is the soln for enterprise logging.
• Rsyslog need the rsyslog service to be running
• The main configuration file is **/etc/rsyslog.conf**
• snap-in files can be placed in **/etc/rsyslog.d/**
• Each logger line contains three items
facility: the specific facility that the log is created for (invented 1980's)
severity: the severity from which should be logged (that can go from debug which is the most verbose to emergency which is only writing logs is a situation arises where system continuity is compromised.)
destination : the file or other destination the log should be written to
• Log files normally are in **/var/log**
• Use the logger command to write messages to rsyslog manually

Understanding Facilities
• **rsyslogd** is and must be backward compatible with the archaic syslog service
• In syslogm a fixed number of facilities was defined, like kern, authpriv, cron and much more
• To work with services that don't have their own facility local {0..7} can be used
• Because of the lack of facilities, some services take care of their own logging and don't use rsyslog

• (Rsyslog provides an option to configure a centralized log service which is a consolidated location where u can read everything about what is going on your servers.)

- **- cd/etc**
- **- cd rsyslog.d/**
- **- ls**
- **- cd ..**
- **- vim rsyslog.conf**
- **(Rules what should be logged and where should be logged)**
- (**Kern.*** is going to /dev/console which is a device that make sure that messages that are logged to /dev/console show up on the console
- * is everything I.e. everything generate by kernel goes to console)
- (most of the log messages are generated by /var/log/messages ;so everything with a severity of info and higher nothing but that is generated by mail or authpriv or cron is going to /var/log/messages)
- **(authpriv.*** shows the authentication related messages)
- **(mail.*** about email handling; "- "means the log line can be stored in a buffer and forced all together through maillog to make write more efficient)
- **(*.emerg** ; omusrmsg is a output module OM for output module which make sure that message is written to the all users , our syslog is a modular)
- **(local7.*** can be configured in local services so somewhere in systemd)

LINUX 12.3 Understanding anacron

- cron.daily, monthly don't have a specific takes indication. Anacron is taking care for it
- **Anacron** is a service behind cron that tales care jobs are executed on a regular basis , but not at a specific time.
- It takes care of the jobs in **/etc/cron.{hourly, daily, weekly, monthly}**
- Configuration is in **/etc/anacrontab**

- **vim /etc/anaconda**
- Anacron by run-parts make sure of staring cron.{hourly, daily, weekly, monthly}
- **cd ../cron.d**
- **cat 0hourly**

- Helpful for backup jobs even if server is down anacron will be running it in specified time.

LINUX 12.4 Scheduling with cron

- Choose the option you want to use
- **crontab -e** as a specific user
- Create a cron file in**/etc/cron.d**
- Use the time specification: ***/10 4 11 12 1-5**
- . **Minute (*/10)**
- **. hour(4)**
- **. day of month (11)**
- **. month (12)**
- **. day of week (1-5)**
- Note that cron does not have **STDOUT**
- So any command that writes to standard out is going to fail so echo can't be used in a cron job but we can use logger which will writes to the syslog.

- **date**
- **crontab -e**
- **18 16 * * * logger hello**
- **tail /var/log/messages**
- **less /var/log/messages**
- **man 5 crontab**

LINUX 12.5 Scheduling Tasks with Systemd Timers

- **Systemd timers** also allow for scheduling jobs on regular basis , cron however is still the standard.
- **read man 7 systemd-timer** for more info about systemd timers
- **read man 7 systemd-time** for specification of the time format to be used.

- **cd /usr/lib/systemd/system**
- **ls *timer**
- **ls fstrim***
- **vim fstrim.timer**
- **vim fstrim.service**
- **systemctl status fstrim.service**
- **systemctl status fstrim.timer**
- **systemctl start fstrim.timer**
- **systemctl enable fstrim.timer**
- **systemctl restart fstrim.timer**

LINUX 12.6 Using at

- **at** was developed to run the job at specific time.
- The **atd** service must be running to run once-only jobs using at
- Use **at <time>** to schedule a job
- type one or more job specifications in the at interactive shell
- Use **Ctrl-D** to close this shell
- Use **atq** for a list of jobs currently scheduled
- Use **atrm** to remove jobs from the list

- **systemctl status atd**
- **at teatime**
- **logger have a cup of tea**
- **Mail -s hello root < .**
- **Ctrl -D**
- **atq** it show everything which is scheduled
- **atrm 1** it will remove the at 1
- **atq**

12.7 Managing temporary files

- Temporary in RHEL 8 are managed by systemd-tmpfiles
- The /usr/lib/tmpfiles.d directory manages settings for creating,deleting and cleaning up of temporary files.
- The systemd-tmpfiles-clean.timer unit can be configured to automatically clean up temporary files
- It triggers the systemd-tmpfiles-clean.service
- This service runs systemd-tmpfiles --clean
- (Service specifies what to do and timer specifies when to do)
- The /usr/lib/tmpfiles.d/tmp.conf file contains settings for the automatic tmp file cleanup
- When making modifications, copy the file to /etc/tmpfiles.d
- (In RHEL8, the default configuration is a user lib something and related administrator created configuration is an etc.)
- After making modifications to this file , use systemd-tmpfiles --clean /etc/tmpfiles.d/tmp.conf to ensure the file does not contain any errors.

- In tmp.conf you will find lines specifying which directory to monitor which permissions are set on that directory which owners and after how many days of being used the tmp files will be removed.
- Different actions can be performed on the directories and files that are managed.
- Consult man tmpfiles.d for more

- systemctl cat systemd-tmpfiles-clean.timer
- (On Boot sec which specifies the amount of time after boot that this unit will run .
- On unit active sec which specifies the interval between runs. If you want to modify u should use systemctl daemon reload to tell systemd about it)
- cp /ysr/lib/tmpfiles.d/tmp.conf /etc/tmfiles.d/
- vim etc/tmpfiles.d/tmp.conf
- man tmpfiles.d
- systemd-tmpfiles --clean etc/tmpfiles.d/tmp.conf
- vim /etc/tmpfiles.d/mytemp.conf
- d /run/mytemp 0700 root root 30s
- cd /etc/tmpfiles.d/
- ls
- cd /usr/lib/tmpfiles.d/
- ls
- systemd-tmpfiles --create /etc/tmpfiles.d/mytemp.conf
- touch /run/mytemp/myfile
- ls /run/mytemp/myfile
- sleep 30
- ls /run/mytemp/myfile
- systemd-tmpfiles --clean /etc/tmpfiles.d/mytemp.conf

Thank You